WHAT IS CLAIMED IS:

1. A system comprising:

a network coupling a first subsystem and a second subsystem;

the first subsystem comprising a first processing subsystem providing logic

(a) for processing of streaming data packets, according to defined rules for processing streaming data packets, and

(b) for generation and selectively sending of security tag vectors; and

a second subsystem comprising a second processing subsystem

(a) for sending the streaming data packets to the first subsystem,

(b) for receiving the security tag vectors, and

(c) for providing logic for validating the received security tag vectors responsive to a defined validation logic.

2. The system as in Claim 1, wherein the second subsystem further comprises a transmission controller for stopping the sending of the streaming data packets responsive to the defined validation logic of the selectively transmitted security tag vectors.

3. The system as in Claim 1, wherein the second subsystem is further comprised of a forwarding controller for stopping the forwarding of the streaming data packets responsive to the defined validation logic of the selectively transmitted security tag vectors.

4. The system as in Claim 1, further comprising a defined sequence of decryption keys; and

wherein the defined sequence of decryption keys are sent from the second subsystem to first subsystem responsive to the defined validation logic of the selectively transmitted security tag vectors.

5. The system as in Claim 1, wherein the processing of streaming data packets is further comprised of processing logic; and

wherein the processing logic is further comprised of at least one of: a privileges table, a privileges decision-tree, pseudo random rendering logic, a streaming data packet header processing privileges decision-tree, a security tag processing logic, a streaming

data packet identification processing logic, a secure time-stamp processing logic, a processing of streaming data packets with secure time-stamps, watermarking information processing, fingerprinting information processing, stenographic information processing, data embedding information processing, digital signature information processing, and a processing of streaming data packets with secure time-stamps that is responsive to UTC (coordinated universal time).

6. The system as in Claim 1, wherein the processing of streaming data packet is constructed with codes and parameters in accordance with XrML (Extensible Rights Markup Language).

7. The system as in Claim 1, wherein at least one of: the logic of the first processing subsystem, the defined rules for processing, and the security tag vector generation are further characterized as responsive to a at least one of:

a predefined schedule, a secure time-stamp, renewable codes and parameters, updated codes and parameters, a predefined schedule received from the second subsystem, a secure time-stamp received from the second subsystem, renewable codes and parameters received from the second subsystem, updated codes and parameters received from the second subsystem, a predefined schedule received from a third subsystem, a secure time-stamp received from a third subsystem, renewable codes and parameters received from a third subsystem, and updated codes and parameters received from a third subsystem.

8. The system as in Claim 7, wherein at least one of: selected parts of the logic of the first processing subsystem, selected parts of the defined rules for processing, selected parts of the security tag vector generation, selected parts of the renewable codes and parameters, and selected parts of the updated codes and parameters are provided from an external storage medium.

9. The system as in Claim 8, wherein the external storage medium is at least one of: a smart card, a tamper-proof device, obfuscated storage, hidden storage, encrypted data storage, removable storage, a token card, and a metro card.

10. The system as in Claim 1, wherein at least one of: selected parts of the first logic, selected parts of the defined rules for processing, and selected parts of the security tag vector generation define a plurality of logic modules that are interlocked for streaming data packet processing together with the security tag vector generation.

11. The system as in Claim 10, wherein interlocked is further characterized in that each respective one of the plurality of logic modules is associated with a respective one of a plurality of defined subtasks;

wherein the combined plurality of defined subtasks defines the said selected parts of the logic of the first processing subsystem, said selected parts of the defined rules and said selected parts of the security tag vector generation, and

wherein all of the logic modules are required to properly perform the respective defined subtask to provide the said selected parts of the logic of the first processing subsystem, said selected parts of the defined rules and said selected parts of the security tag vector generation.

12. The system as in Claim 1, further comprises an update controller providing at least one of: updated codes, updated parameters, update decryption codes, update decryption keys, update rendering codes, update playing codes, and updated secure time stamp to the first subsystem.

13. The system as in Claim 12, further comprises a security management server (SMS) for providing update information to the update controller.

14. The system as in Claim 1, further comprises a renewable controller providing at least one of: updated codes, updated parameters, update decryption codes, update decryption keys, update rendering codes, update playing codes, and updated secure time stamp to the first subsystem.

15. The system as in Claim 14, further comprises a security management server (SMS) providing renewable information to the renewable controller.

16. The system as in Claim 1, wherein the first subsystem is further comprised of cryptographic modules; and

wherein the cryptographic modules provide for at least one of:

program authentication, user authentication, cryptographic authentication, application authentication, encryption, a secure time-stamp, a digital signature, watermarking information, IPSec (IP Security) functionality, TLS (Transport Layer Security) functionality, and SSL (Secure Sockets Layer) functionality.

17. The system as in Claim 1, wherein the second subsystem is further comprised of validation modules; and

wherein the validation modules further provide for at least one of: program authentication checking, user authentication checking, cryptographic authentication checking, application authentication checking, decryption, a secure time-stamp, a digital signature validation, validation of watermarking information, IPSec (IP Security) functionality, TLS (Transport Layer Security) functionality, and SSL (Secure Sockets Layer) functionality.

18. The system as in Claim 1, wherein the first subsystem further includes a media player.

19. The system as in Claim 1, wherein the first subsystem further is included within a media player.

20. The system as in Claim 18, wherein the media player is directly attached to at least one of:

a video display, a TV display, a computer monitor, a handheld display, an audio speaker, a stereo audio system, a digital output system, an analog output system, and a media play buffer.

21. The system as in Claim 18, wherein the media player performs at least one of:

deleting streaming data packets after processing, deleting streaming data packets within a predefined time interval after processing, deleting streaming data packets after a

defined number of times of processing, preventing copying of the streaming data packets, preventing printing of the streaming data packets, preventing sending of the streaming data packets, encrypting video rendering of content received in the streaming data packets, pseudo random video rendering of content received in the streaming data packets, encrypting video rendering of content stored in the first subsystem, and pseudo random video rendering of content stored in the first subsystem.

22. The system as in Claim 18, wherein the media player operates in accordance with at least one of:

XrML (Extensible Rights Markup Language) specifications, trusted computing specifications, trusted computing based principles, validation of watermarking information, IPSec (IP Security) functionality, TLS (Transport Layer Security) functionality, and SSL (Secure Sockets Layer) functionality.

23. The system as in Claim 1, wherein the second subsystem further includes a media server.

24. The system as in Claim 23, wherein the media server operation is responsive to the security tag vectors sent from the first subsystem.

25. The system as in Claim 1, wherein there is a plurality of the first subsystems, each coupled to the network and receiving streaming data packets from the second subsystem.

26. The system as in Claim 25, wherein the second subsystem encodes the respective streaming data packets responsive to validating the respective received security tag vectors from the respective one of the plurality of the first subsystems.

27. The system as in Claim 26, wherein the streaming data packets are encoded such that any of the first subsystems in which the validating of their security tag vectors fails will not thereafter be able to further decode the streaming data packets.

28. The system as in Claim 26, wherein the streaming data packets are encrypted by the second subsystem by using a group encryption scheme such that any of the first subsystems in which the validating of their security tag vectors fails will not be able to further decode the streaming data packets.

29. The system as in Claim 26, wherein the streaming data packets are sent using at least one of:

Multicast, IP (Internet Protocol) Multicast, Secure IP Multicast, Group Key Management Architecture, Multi-Party Non-Repudiation Protocol, Group Communications, and Secure Group Communications.

30. The system as in Claim 1, wherein there is a plurality of second subsystems coupled to the network, each sending a respective plurality of streaming data packets to the first subsystem.

31. The system as in Claim 30, wherein the first subsystem sends security tag vectors to the plurality of second subsystems for validation.

32. The method as in Claim 1, wherein the first subsystem is at least one of:

a wireless device, a handheld device, a Wi-Fi device, a device operating in accordance with IEEE 802.11 family of standards, a device operating in accordance with IEEE 802.15, a 2.5G cellular telephone, a 3G cellular telephone, a 4G cellular telephone, a 5G cellular telephone, a personal computer, a set-top box, a device operating in accordance with UMTS (Universal Mobile Telephone System), and a device operating in accordance to the IEEE 802.3 family of standards.

33. The system as in Claim 1, wherein the second subsystem further comprises encryption logic for encrypting streaming data packets prior to sending them; and

wherein the first subsystem logic for processing of streaming data packets further comprises logic for decrypting the streaming data packets.

34. The system as in Claim 33, wherein the second subsystem provides an encryption key to the first subsystem logic.

35. The system as Claim 34, wherein the encryption key is provided at least one of: periodically, at random times, at predefined time intervals, responsive to validating the received security tag vectors, and at predefined times derived from coordinated universal time (UTC).

36. The system as in Claim 33, wherein the first subsystem further comprises logic for generating and sending encryption keys to the second subsystem; and

wherein the second subsystem uses the encryption keys for encrypting the streaming data packets prior to sending them.

37. A method for authenticated operation on data flows between at least a first computing element and a second computing element, the method comprising:

receiving the data from the first computing element;

processing the data and generating security tag vectors in the first computing element;

sending the security tag vectors to the second computing element; and

validating the security tag vectors in the second computing element to determine compliant communication of the data.

38. The method as in Claim 37, further comprising:

providing a plurality of software logic modules and parameters operable stand-alone to provide a respective plurality of subtask functions as part of the first computing element;

providing secure integration of the plurality of software logic modules and parameters to provide a combined functionality as part of the first computing element;

interlocking the plurality of software logic modules and parameters into a single logic program as part of the first computing element; and

providing the combined functionality only when the plurality of subtask functions are executed responsive to the single logic program as part of the first computing element.

39. The method as in Claim 38, further comprising:

producing a pseudo-random sequence of security tag vectors by the first computing element.

40. The method as in Claim 39, further comprising:

producing the pseudo-random sequence of security tag vectors utilizing computation by at least one of:

applying a pseudo-random generator, applying a pseudo-random function, applying a cryptographic function, applying an encryption function, applying a scrambling subroutine, applying an authentication function, applying a digital signing function, applying a cryptographic hash function, applying a subroutine, applying a computational logic module, applying a symmetric cryptography function, applying an asymmetric cryptography function, employing a cryptographic key, employing a cryptographic seed, employing an encrypted software, employing an obfuscated software, employing a hidden program, employing watermarking information, employing fingerprinting information, employing digital signature information, employing logic with a set of parameters, employing a hardware module, employing a smart card, employing a portable device, and employing a distributed protocol.

41. The method as in Claim 37, further comprising:

producing a pseudo-random sequence of security tag vectors responsive to the processing of data in the first computing element.

42. The method as in Claim 37, further comprising:

sending the data from the second computing element, and

wherein upon failing the validating of the security tag vectors by the second computing element the sending of data is stopped.

43. The method as in Claim 38, further comprising:

a smart card that is part of the first computing element, and

wherein selected modules and parameters of the plurality of software logic modules and parameters reside on the smart card.

44. The method as in Claim 38, wherein selected ones of the parameters are at least one of: an encryption key, a decryption key, and an authentication parameter.

45. The method as in Claim 38, further comprising:

renewing selected modules and parameters of the plurality of software logic modules and parameters.

46. The method as in Claim 38, further comprising:

replacing selected modules and parameters of the plurality of software logic modules and parameters.

47. The method as in 45, wherein the renewing is performed in at least one of:

periodically, at random times, at predefined times, at predefined times derived from coordinated universal time (UTC), responsive to receiving data by the first computing element, responsive to sending the security tag vectors, and responsive to sending data by the second computing element.

48. The method as in Claim 38, further comprising:

erasing data from memory by the single logic program as part of the first computing element.

49. The method as in Claim 48, wherein memory is at least one of: solid state, a magnetic storage device, and an optical storage device.

50. The method as in Claim 48, wherein erasing data is performed responsive to at least one of:

after predefined time, after the data was output to an output device, and after the data was output predefined number of times to an output device.

51. The method as in Claim 37, wherein the first computing element is in at least one of:

a wireless device, a handheld device, a Wi-Fi device, a device operating in accordance with IEEE 802.11, a device operating in accordance with IEEE 802.15, 2.5G cellular telephone, a 3G cellular telephone, a 4G cellular telephone, a 5G cellular telephone, a personal computer, a set-top box, a device operating in accordance with UMTS (Universal Mobile Telephone System), and a device operating in accordance with IEEE 802.3 family of standards.

52. A method of providing content protection in streaming data packets, the method comprising:

defining within a first subsystem defined rules for processing;

receiving the streaming data packets in the first subsystem;

processing of the streaming data packets in the first subsystem according to defined rules for processing;

generating security tag vectors responsive to the defined rules for processing;

sending the security tag vectors from the first subsystem to a second subsystem;

providing defined validation logic in the second subsystem;

processing, in the second subsystem, the received security tag vectors, responsive to the defined validation logic to provide respective validated security tag vectors; and

processing in the second subsystem the validated security tag vectors and the received security tag vectors to determine compliant communication of the streaming data packets to the first subsystem from a third subsystem.

53. The method as in claim 52, wherein the content is representative of at least one of: a movie, a book, a music piece, concert, a 3D movie, a sport event;

wherein the content is divided into predefined number of parts, and

wherein each subset of parts is associated with a decryption key.

54. A communication method for authentication of communications of data packets, the method comprising:

defining rules of processing;

generating security tag vectors responsive to the rules of processing and the data packets;

transmitting data packets from a second subsystem to a first subsystem;

receiving the transmitted streaming data packets for processing in the first subsystem;

sending respective ones of the security tag vectors from the first subsystem to the second subsystem, responsive to the data packets and the rules of processing; and

processing the received security tag vectors in the second subsystem to assure that the processing in the first subsystem is compliant with the defined rules of processing.

55. The method as in Claim 54, further comprising:

validating the received security tag vectors in the second subsystem responsive to a second rules of processing; and

interlocking the validating of the received security tag vectors in the second subsystem with the defining of the rules of processing in the first subsystem to assure compliant communication.

56. A system for providing remotely authenticated operations, the system comprising:

a tag generator operating from an initial generator state to generate a sequence of security tag vectors responsive to a sequence of content processing steps;

means providing for transmission of the sequence of security tag vectors;

a tag verifier operating from an initial verification state to generate a sequence of comparison security tag vectors for selective comparison to sequence of the security tag vectors; and

means for coordinating the initial generator state and the initial verifier state prior to the sequence of content processing steps,

wherein the tag verifier selectively provides valid comparison tags responsive to the means for coordinating.

57. The system as in Claim 56, wherein the tag generator includes a sequence number as part of the security tag vector.

58. The system as in Claim 57, wherein the tag verifier generates a comparison sequence number for selective comparison to the sequence number that is part of the security tag vector.

59. The system as in Claim 57, wherein the sequence number is used for at least detecting a security tag vector loss.

60. The system as in Claim 56, wherein the tag generator provides a secure time-stamp as part of the security tag vector.

61. The system as in Claim 60, wherein the tag verifier generates a comparison secure time-stamp for selective comparison to the secure time-stamp that is part of the security tag vector.

62. The system as in Claim 56, further comprising:

means for remotely downloading of codes and parameters.

63. The system as in Claim 62, wherein the codes and parameters are used to perform at least one of: processing streaming data packets, and generating security tag vectors.

64. A system for providing secure integration of separate logic modules to provide a combined functionality, the system comprising:

a first processing subsystem

(a) for processing of streaming data packets, responsive to defined rules for processing streaming data packets, and

(b) for generation and selectively sending of security tag vectors;

wherein the first processing subsystem is further comprised of a plurality of software logic modules each operable stand-alone to provide a respective one of a plurality of subtask functions; and

a transformation controller for interlocking the plurality of software logic modules into a single logic program;

wherein the combined functionality is only provided when the plurality of subtask functions are executed responsive to the single logic program.

65. The system as in Claim 64, wherein the single logic program is written to be immune to reverse generation.

66. The system as in Claim 64, wherein one of the software logic modules provides a cryptographic function for producing a pseudo-random sequence of security tags.

67. The system as in Claim 66, wherein producing pseudo-random sequence of security tags utilizes computation by at least one of:

applying a pseudo-random generator, applying a pseudo-random function, applying a cryptographic function, applying an encryption function, applying a scrambling subroutine, applying an authentication function, applying a digital signing function, applying a cryptographic hash function, applying a subroutine, applying a computational logic module, applying a symmetric cryptography function, applying an asymmetric cryptography function, employing a cryptographic key, employing a cryptographic seed, employing an encrypted software, employing an obfuscated software, employing a hidden program, employing logic with a set of parameters, employing a hardware module, employing a smart card, employing a portable device, and employing a distributed protocol.

68. The system as in Claim 64, wherein one of the software logic modules provides logic to process the content of received streaming data packets.

69. The system as in Claim 68, wherein logic to process the content of received data packets performs at least one of:

video rendering of the content on a video display, playing the content via audio speakers, displaying the content on an e-book output device, outputting the content to an output device, and outputting the content to an analog output device.

70. The system as in Claim 64, wherein one of the software logic modules provides rules of playing audio and video content.

71. The system as in Claim 70, wherein the rules of playing audio and video content ensure at least one of:

the content is not printed; the content is not sent to a third party; the content is destroyed after being displayed on a video monitor; the content is being destroyed after being played via an audio speakers; the content is erased from all memory storage devices after being displayed on a video monitor; the content is erased from all memory storage devices after being played via an audio speakers; the content is erased from all memory storage devices after being used via an e-book output device; the content is erased from all memory storage devices after a predefined time interval; the content is erased from all memory storage devices at a time defined time by coordinated universal time (UTC); the content is used in accordance with rights defined using XrML (Extensible Rights Markup Language) specifications; the content is used in accordance with trusted computing specifications; the content is used in accordance with trusted computing based principles; and the content is used in accordance with at least one of the following: watermarking information, stenographic information, fingerprinting information, embedded data and digital signature information.

72. The system as in Claim 70, wherein at least one of the rules of content processing determines a renewable software for content processing.

73. The system as in Claim 72, wherein the renewable software for content processing is least one of:

number of times the content can be displayed, number of times the content can be played, a time signal, a UTC time signal, a digitally signed time signal, a software element, a predefined task, a code for processing content signature, and a code for watermarking the content.

74. The system as in Claim 72, wherein the renewable software for content processing is obtained from at least one of:

a second subsystem, a second computing element, predefined logic, an external rule controller, a security management system, via a network interface, a network appliance, a server, a network management system, a firewall, local computation, a smart card device, and a portable device.

75. The system as in Claim 64, wherein one of the software logic modules provides a cryptographic function for producing a pseudo-random sequence of security tag vectors; and

wherein one of the software logic modules provides logic to process and play audio and video content.

76. The system as in Claim 64, wherein one of the software logic modules provides a cryptographic function for verifying at least one of: watermarking, embedded data and fingerprinting.

77. The system as in Claim 64, further comprising:

a source of interlocking parameters, and

wherein the transformation controller is further comprised of means for combining the software logic modules according to defined interlocking logic responsive to the interlocking parameters.

78. The system as in Claim 77, wherein the source of interlocking parameters is generated by at least one of:

a random source, a cryptographic key, and a defined table and location in memory.

79. The system as in Claim 77, wherein the transformation controller determines an intermixture of the subtask functions of the plurality of software logic modules into the single program to provide the combined functionality.

80. The system as in Claim 79, wherein the intermixture can be provided in a defined plurality of different ways; and

wherein each of the different ways provides a different one of the single program providing the combined functionality.

81. The system as in Claim 80, wherein the intermixture is further comprised of at least one of:

obfuscation, encryption, replication, adding dummy code, addition of redundant control, renaming of variables, splitting a procedure into multiple sub-procedure, dictionary transformation, compilation, interpretation, cryptographic transformation, digital signing, and scrambling.

82. The system as in Claim 81, wherein the replication is comprised of repetitions of the software logic modules into an oversize program comprising the single program therein embedded.

83. The system as in Claim 82, wherein each repetition is made active separately to define an active single program within the oversize program which acts as the single program.

84. The system as in Claim 64, wherein the transformation controller further generates external software modules for linked operation with the single program required for the combined functionality.

85. The system as in Claim 84, further comprising:

means for transmitting the external software modules to separate computing subsystems, and

wherein the external software modules are executed in the separate computing subsystems to provide at least one of: update information and renewable information coupled to the single logic program.

86. The system as in Claim 85, wherein the means for transmitting utilizes at least one of: encryption, authentication, and digital signing.

87. The system as in Claim 85, wherein the update information is at least one of: change data, change executable code, change pattern, change order and pseudo-change of dummy code.

88. The system as in Claim 85, wherein the renewable information is at least one of:

renewable content processing, time signal, a UTC time signal, a digitally signed time signal, a digital cache for transmission of trusted content processing, and a cryptographic key for marking trusted content processing.

89. The system as in Claim 64, further comprising:

means for transmitting the single logic program to a primary computing system.

90. The system as in Claim 89, wherein the means for transmitting utilizes at least one of: encryption, authentication, watermarking, and digital signing.

91. The system as in Claim 64, wherein security verification information is generated by the transformation controller, for utilization by separate security tag verification logic in a separate subsystem which validates the security tag.

92. The system as in Claim 64, wherein one of the software logic modules provides security services.

93. The system as in Claim 92, wherein the security services provide for at least one of:

user authentication, user sign-on, data packet authentication, user login, applying a user's cryptographic key, applying an organization's cryptographic key, group encryption, watermarking validation, and digital signing.

94. The system as in Claim 92, wherein the security services further provide for applying cryptographic transformations based on keys belonging to a primary computing system.

95. The system as in claim 94, wherein the primary computing system provides for execution of the single logic program.

96. A method of providing controlled signaling, the method comprising:

providing defined rules of at least one of: transmission, forwarding, and operation;

processing streaming data packets in accordance with the defined rules;

generating a security tag vector responsive to validating the processing in accordance with the defined rules; and

constructing a signal responsive to computing with the security tag vector.

97. The method as in Claim 96, further comprising:

transmitting the signal onto a communications path in accordance with the constructing.

98. The method as in Claim 97, further comprising:

receiving the signal from the communications path; and

validating the signal responsive to the computing with the security tag vector.

99. The method as in Claim 96, further comprising:

receiving at least some of the defined rules of at least one of: transmission, forwarding, and operation, from a separate rules controller.

100. The method as in Claim 96, further comprising:

determining a renewable software module for at least one of: transmission, forwarding, and operation responsive to at least one of the defined rules of at least one of: transmission, forwarding, and operation.

101. The method as in Claim 96, wherein the generating comprises at least one of:

applying a pseudo-random generator, applying a pseudo-random function, applying a cryptographic function, applying an encryption function, applying a scrambling subroutine, applying an authentication function, applying a digital signing function, applying a cryptographic hash function, applying a subroutine, applying a computational logic module, applying a symmetric cryptography function, applying an asymmetric cryptography function, employing a cryptographic key, employing a cryptographic seed, employing an encrypted software, employing an obfuscated software, employing a hidden program, employing logic with a set of parameters, employing a hardware module, employing a smart card, employing a portable device, and employing a distributed protocol.

102. The method as in Claim 97, wherein the communications path connects a network interface with at least one of:

a wireless device, handheld device, a Wi-Fi device, a device operating in accordance with IEEE 802.11, a device operating in accordance with IEEE 802.15, a 2.5G cellular telephone, a 3G cellular telephone, a 4G cellular telephone, a 5G cellular telephone, a personal computer, a set-top box, a device operating in accordance with UMTS (Universal Mobile Telephone System), and a device operating in accordance to IEEE 802.3 family of standards.